AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently amended) A computer-implemented system that test loads a server comprising:

a plurality of simulated users utilized to test load a server; and

a dynamic load adjustor component that dynamically adjusts ~~user characteristics~~ at least one characteristic of at least one of the simulated users based at least in part on a browser type related to the simulated user, for distribution thereof as a percentage of total requests sent to [[a]] the server being load tested.

2. (Previously Presented) The system of claim 1, further comprising a profile characteristic data store that supplies the dynamic load adjustor component with weighting for a characteristic defined in a user profile.

3. (Original) The system of claim 2, the dynamic load adjustor component further comprises a weighting designator that randomly assigns to users characteristics based on weightings defined in the user profile.

4. (Previously Presented) The system of claim 2, the characteristic comprises at least one of: network connections, browser types, and load patterns.

5. (Previously Presented) The system of claim 2, the characteristic statistically determined based on web log records.

6. (Previously Presented) The system of claim 2, the characteristic predetermined in a single user profile.

7. (Currently Amended) The system of claim 1, further comprising a load coordinator component that adjusts an intensity of a load test based on a current distribution of simulated users entering and leaving the server relative to a desired test load.

8. (Original) The system of claim 1, further comprising an artificial intelligence component.

9. (Previously Presented) The system of claim 1, further comprising a closed loop control to enable a continual and sustained rate of requests to the server.

10. (Currently Amended) A machine-implemented system that stresses a server, comprising:

an execution engine that generates a scenario that loads the server *via* a plurality of simulated users, the plurality of simulated users dynamically adjusted based on predetermined weightings of a user profile related to at least one of the simulated users having weighted characteristics that comprises at least a browser type therein, wherein the scenario distributes user characteristics as a percentage of total requests.

11. (Original) The system of claim 10, the scenario comprises at least one of a test mix and a load profile.

12. (Previously Presented) The system of claim 10, further comprising a control input that adjusts rate of requests loaded onto the server.

13. (Previously Presented) The system of claim 10, further comprising a queuing mechanism that retrieves and sorts requests to be sent to the server.

14. (Previously Presented) The system of claim 10, further comprising a scheduler that determines number of requests to be generated for an upcoming period.

15. (Previously Presented) The system of claim 10, the requests sorted according to a time function for execution.

16. (Currently amended) A computer-implemented method for load testing a server comprising:

assigning weights to user characteristics in a user profile;

dynamically adjusting the user characteristics based on one or more browser types during the testing of the server; and

distributing the user characteristics as a percentage of total simulated user requests sent to the server.

17. (Previously Presented) The method of claim 16, further comprising comparing a current load on the server with a desired load.

18. (Previously Presented) The method of claim 17, further comprising creating a new user if the current load falls below a desired load.

19. (Previously Presented) The method of claim 17, further comprising reducing the current load by one upon ending an iteration, if the current load rises above the desired load.

20. (Previously Presented) The method of claim 16, further comprising controlling a rate of loading *via* a feedback loop control.

21. (Currently amended) A machine-implemented system for test loading a server comprising:

means for dynamically adjusting ~~user~~ characteristics of a simulated user while loading the server; and

means for distributing the simulated user characteristics as a percentage of total requests sent to the server, each user characteristic including at least a browser type.